
POINT LOCATION IN DISCONNECTED PLANAR SUBDIVISIONS

Prosenjit Bose, Luc Devroye, Karim Douïeb, Vida Dujmović, James King, and Pat Morin

January 15, 2010

ABSTRACT. Let G be a (possibly disconnected) planar subdivision and let D be a probability measure over \mathbb{R}^2 . The current paper shows how to preprocess (G, D) into an $O(n)$ size data structure that can answer planar point location queries over G . The expected query time of this data structure, for a query point drawn according to D , is $O(H + 1)$, where H is a lower bound on the expected query time of any linear decision tree for point location in G . This extends the results of Collette *et al.* (2008, 2009) from connected planar subdivisions to disconnected planar subdivisions. A version of this structure, when combined with existing results on succinct point location, provides a succinct distribution-sensitive point location structure.

1 Introduction

Planar point location is the classic search problem in computational geometry. The problem asks us to preprocess a planar subdivision G so that we can quickly test, for any query point p , which face of G contains p . Optimal, $O(n)$ space, $O(\log n)$ query time structures for the point location problem have been known for over 25 years [17, 22, 24, 26], the precise constants achievable in the query time are well-understood [1], several results exist for distribution-sensitive query times [3, 4, 5, 6, 7, 12, 13, 20, 21], and sublogarithmic query time data structures exist for transdichotomous models of computation [9, 10, 25].

The most recent work in the distribution-sensitive setting is by Collette *et al.* [12] who give an $O(n)$ space data structure that preprocesses a connected planar subdivision G and a probability measure D over \mathbb{R}^2 such that a point location query in G can be answered in $O(H + 1)$ expected time. Here H is a lower-bound on the expected time required by any linear decision tree for answering queries on G that are drawn according to D . The expected number of point-line comparisons needed to answer a query using their data structure is $H + O(H^{2/3} + 1)$. Their work, which generalizes (and uses) a similar result for triangulations [7], leaves open the problem of what to do when G is disconnected. Disconnected planar subdivisions occur quite frequently in areas like geographic information systems and cartography, where disconnected regions occur naturally. (See Figure 1 for example).

In the current paper we show that, for a (possibly disconnected) planar subdivision G , a very different approach can be used to obtain an expected query time of $O(H + 1)$. Essentially, the problem can be solved by building a $o(n)$ -sized data structure for answering the easy-to-answer queries efficiently and passing all other (hard-to-answer) queries on to



Figure 1: A disconnected planar subdivision that occurs in the context of cartography.

any of the classic $O(n)$ space $O(\log n)$ query time data structures for planar point location. As a corollary, we obtain a succinct distribution-sensitive data structure for point location in (possibly-disconnected) subdivisions. This data structure stores only a permutation of the vertices of the subdivision plus an additional $o(n)$ bits.

2 Preliminaries

Throughout this paper, we assume an underlying probability measure D over \mathbb{R}^2 . All expectations and probabilities are (implicitly) with respect to D . For any subset $X \subseteq \mathbb{R}^2$, $\Pr(X)$ refers to $D(X)$. We use the notation $D|_X$ to denote the distribution D conditioned on X , i.e., $D|_X(Y) = \Pr(Y | X) = \Pr(X \cap Y) / \Pr(X)$ for all $Y \subseteq \mathbb{R}^2$. If Δ is a partition of \mathbb{R}^2 , then the *entropy* of Δ , denoted $H(\Delta)$ is

$$H(\Delta) = \sum_{t \in \Delta} \Pr(t) \log(1 / \Pr(t)) .$$

The probability measure D is used as an input to our algorithms. We assume that the algorithm has access to D through two oracles. The first oracle allows, for any triangle t , to determine $\Pr(t)$ in constant time. The second oracle, for any triangle t , allows the algorithm to draw a point p according to $D|_t$ in constant time.

A *linear decision tree* for point location over G is a rooted binary tree in which each internal node v is labelled with a linear inequality $a_v x + b_v y + c_v > 0$, and each leaf ℓ is labelled with a face of G . A query point $p = (x, y)$ follows a root-to-leaf path, proceeding to the left child of v if it satisfies the inequality and the right child of v if it does not. A

linear decision tree is *for point location in G* if, for every $p \in \mathbb{R}^2$ the path for p ends at a leaf labelled with the face of G that contains p . In the case where p lies on an edge or vertex of G , the label can be any of the faces of G incident on that edge or vertex. The (*expected*) *cost* of a linear decision tree is the expected depth of the leaf reached when p is drawn according to the probability measure D .

3 The Data Structure

In this section we describe our data structure for point location in disconnected planar subdivisions. The first tool we use is simplicial partitions, from the field of geometric range searching:

Theorem 1 (Matoušek 1992). *There exists a universal constant c such that, for any set S of m points in \mathbb{R}^2 and any $r \in \{1, \dots, m\}$, there exists a sequence $\langle \Delta_1, \dots, \Delta_r \rangle$ of closed triangles such that*

1. $S \subseteq \bigcup_{i=1}^r \Delta_i$,
2. $\left| \Delta_i \cap S \setminus \left(\bigcup_{j=1}^{i-1} \Delta_j \right) \right| \leq 2m/r$, and
3. *For any line ℓ , there are at most $cr^{1/2}$ elements of $\{\Delta_1, \dots, \Delta_r\}$ whose interiors intersect ℓ .*

The sequence of triangles $\Delta_1, \dots, \Delta_r$ can be computed in $O(m)$ time.

Note that Part 2 of Theorem 1 is not in the original statement of the theorem, but follows from Matoušek's construction of $\Delta_1, \dots, \Delta_r$ [23]. Restating Theorem 1 in terms of probability distributions, we have:

Theorem 2. *There exists a universal constant c such that, for any probability measure D over \mathbb{R}^2 and any integer $r \geq 1$, there exists a sequence $\langle \Delta_1, \dots, \Delta_r \rangle$ of closed triangles such that*

1. $\Pr \left\{ \bigcup_{i=1}^r \Delta_i \right\} = 1$,
2. $\Pr \left\{ \Delta_i \setminus \left(\bigcup_{j=1}^{i-1} \Delta_j \right) \right\} \leq 3/r$, and
3. *For any line ℓ , there are at most $cr^{1/2}$ elements of $\{\Delta_1, \dots, \Delta_r\}$ whose interiors intersect ℓ .*

The sequence $\Delta_1, \dots, \Delta_r$ of triangles can be computed in $O(r^3 \log r)$ time.

Proof. Assume that $r \geq 2$, otherwise the theorem is trivial. We will draw an i.i.d. sample of $m = \lceil 256r^3 \ln r \rceil$ points from D to form a set S . We use the algorithm from Theorem 1 to build a sequence $\langle \Delta_1, \dots, \Delta_r \rangle$ of triangles satisfying the conditions of Theorem 1. If necessary we replace Δ_r with a triangle that contains the support of D to ensure condition

(1) of this theorem is satisfied. Condition (3) of this theorem is the same as condition (3) of Theorem 1 and is therefore trivially satisfied, though it may be necessary to add 1 to the constant c due to the replacement of Δ_r .

We will prove that, with probability at least $1/2$, the sequence $\langle \Delta_1, \dots, \Delta_r \rangle$ also satisfies condition (2) of this theorem. Our oracles allow us to check in constant time whether this condition is satisfied; we repeat the process until we obtain a partition that does. The runtime for this algorithm will then be geometrically distributed with constant expectation for any constant r .

To denote the incremental differences between the triangles we use

$$\Delta_i^* = \Delta_i \setminus \bigcup_{j=1}^{i-1} \Delta_j .$$

We will use $D_m(A)$ to denote the empirical measure of a set A :

$$D_m(A) \stackrel{\text{def}}{=} \frac{|S \cap A|}{m} .$$

By condition (2) of Theorem 1, we have

$$\sup_{1 \leq i \leq r} D_m(\Delta_i^*) \leq \frac{2}{r} .$$

Now,

$$\begin{aligned} \Pr \left\{ \sup_{1 \leq i \leq r} D(\Delta_i^*) > \frac{3}{r} \right\} &= \Pr \left\{ \bigcup_{1 \leq i \leq r} \left[D(\Delta_i^*) - D_m(\Delta_i^*) > \frac{3}{r} - D_m(\Delta_i^*) \right] \right\} \\ &\leq \Pr \left\{ \bigcup_{1 \leq i \leq r} \left[D(\Delta_i^*) - D_m(\Delta_i^*) > \frac{3}{r} - \frac{2}{r} \right] \right\} \\ &= \Pr \left\{ \sup_{1 \leq i \leq r} (D(\Delta_i^*) - D_m(\Delta_i^*)) > \frac{1}{r} \right\} \\ &\leq \Pr \left\{ \sup_{A \in \mathcal{A}} (D(A) - D_m(A)) > \frac{1}{r} \right\} \end{aligned}$$

where \mathcal{A} are sets formed by taking a closed triangle and subtracting at most $r - 1$ closed triangles from it. The class \mathcal{A} for $r = 1$ is the class of all triangles. It has Vapnik-Chervonenkis dimension at most 7. By Sauer's lemma [27][15, Pages 28–29], the number of subsets of an m -point set that can be obtained by intersections with sets from \mathcal{A} does not exceed $(m + 1)^7$. Assume now general r . Then the number of subsets of an n -point set that can be obtained by intersections with sets from \mathcal{A} does not exceed $(m + 1)^{7r}$, by a simple combinatorial argument. Then, by a version of the Vapnik-Chervonenkis Inequality [28] shown by Devroye [14],

$$\Pr \left\{ \sup_{A \in \mathcal{A}} |D(A) - D_m(A)| \geq t \right\} \leq 4e^{4t+4t^2} (m^2 + 1)^{7r} e^{-2mt^2} ,$$

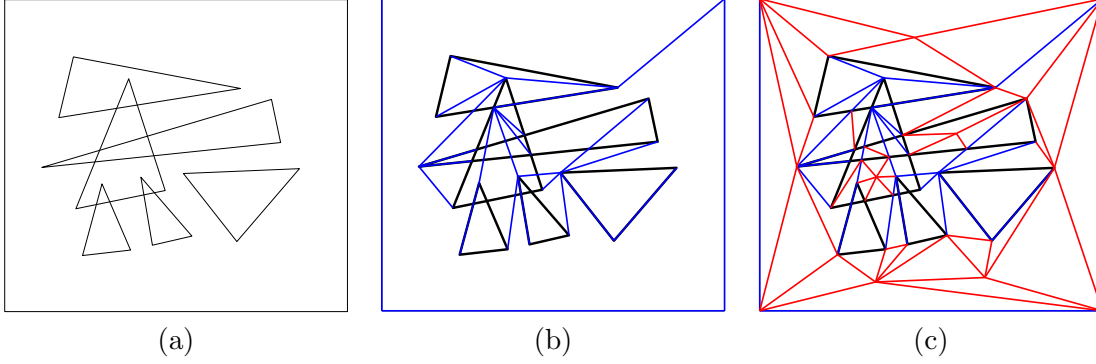


Figure 2: The triangles of a simplicial partition (a) form an arrangement of triangles to which (b) a spanning tree is added, and (c) the faces of the resulting connected subdivision are (Steiner) triangulated to form a Steiner triangulation A .

for any $t > 0$. Thus,

$$\begin{aligned}
 \Pr \left\{ \sup_{1 \leq i \leq r} D(\Delta_i^*) > \frac{3}{r} \right\} &\leq 4e^{4/r+4/r^2} (m^2 + 1)^{7r} e^{-2m/r^2} \\
 &\leq 2^{2+7r} e^8 m^{14r} e^{-2m/r^2} \\
 &\leq \exp \left(31r \ln m - \frac{2m}{r^2} \right).
 \end{aligned}$$

Since we have $m = \lceil 256r^3 \ln r \rceil$, this upper bound is less than $1/2$, as desired. This concludes the proof. \square

Assume, without loss of generality, that all vertices of G and the support of D are contained in the unit square $[0, 1]^2$. This can easily be justified by scaling and translation, so that G is contained in $[0, 1]^2$, and performing 4 point-line comparisons to check that the query point is in $[0, 1]^2$ before using the data structure to answer a query.

We use Theorem 2 to recursively construct a *partition tree* T . Let $\alpha > 0$ be a constant that will be specified below. Refer to Figure 2. At the root of T , we find the sequence of triangles $\Delta = \langle \Delta_1, \dots, \Delta_r \rangle$ and construct the arrangement of triangles in Δ .

Next, we describe how to triangulate this arrangement while maintaining the properties of Theorem 2. Let \mathcal{V} be set of $3r + 4$ points that make up the vertices of the triangles in Δ plus the vertices of a square \square that contains all triangles in Δ . A classic result of Haussler and Welzl [18] proves that V has a spanning tree $T(V)$ such that any line crosses $O(r^{1/2})$ edges of $T(V)$, and this spanning tree can be constructed efficiently [11]. (See Figure 2.b.)

Consider the line segment arrangement L consisting of the union of the edges in $T(V)$, the triangles in Δ , and the edges of \square . Note that any line ℓ intersects $O(r^{1/2})$ edges of the arrangement L ; $O(r^{1/2})$ of these intersections are generated by edges corresponding to edges of $T(V)$ and $O(r^{1/2})$ are generated by edges of triangles in Δ . What remains is to show how to triangulate the faces of L without introducing too many crossings.

By construction, each face F of L , except the outer face, is a (weakly) simple polygon having $O(r)$ vertices and edges on its boundary. By a result of Hershberger and Suri [19], there exists a Steiner triangulation, $A(F)$, of F using $O(r)$ vertices such that any chord of F intersects $O(\log r)$ edges of $A(F)$. We therefore triangulate the arrangement L by triangulating each of its faces in this way. This gives a Steiner triangulation A of L in which any line intersects $O(r^{1/2} \log r)$ edges of A . (See Figure 2.c.)

Next, each face F of A becomes a child of the root of T . If the interior of F is contained in a single face of G then we call F a *terminal leaf* and label F with the face of G that contains it. If the current depth of recursion is greater than $\lfloor \alpha \log_r n \rfloor$ then F becomes a *non-terminal leaf* of T . Otherwise (F intersects two or more faces of G and its depth is small), we recursively apply the same procedure on the distribution $D|_F$ to obtain a partition tree that becomes a child of the root.

This construction defines a tree $T = T(G, D)$ in which each node has $O(r^2)$ children and whose height is at most $\alpha \log_r n$. The number of nodes of T at level i is most $(O(r^2))^i = O(r^{i(2+\epsilon)})$ and therefore the total number of nodes in T is $(O(r^2))^{\alpha \log_r n+1} = O(n^{2\alpha+\epsilon})$, where $\epsilon > 0$ is a decreasing function of r . Note that, for $\alpha < 1/2$ and sufficiently large r , the size of T is $o(n)$.

In addition to the tree T we construct a backup data structure T' that can answer point location queries in G in $O(\log n)$ worst-case time. To answer a query, T and T' are used as follows: We search top-down in T for the query point. If this search ends at a terminal leaf F of T then we report the label at F and the query is complete. Otherwise we use T' to answer the query in $O(\log n)$ time.

4 Analysis

Collette *et al.* [12, 13] show that, up to a lower-order term, the expected number of comparisons performed by the optimal decision tree for point location in G is equal to the entropy of the minimum-entropy Steiner triangulation of G .

Theorem 3 (Collette *et al.* 2008). *Let G be a planar subdivision and let D be a probability measure over \mathbb{R}^2 . Let T^* be a minimum-entropy Steiner triangulation of G and let H^* be the entropy of T^* . Then any linear decision tree for point location in G has expected cost at least $H^* - O(\log H^*)$.*

Thus, our goal is to prove that our query time approximates the entropy of the minimum entropy Steiner triangulation of G . We begin by showing that the partition tree T has small *visiting number* [18].

Lemma 1. *Let $\epsilon > 0$, and let T be the partition tree defined in Section 3 using a value r such that $r > (c \log r)^{1/\epsilon}$ for some (sufficiently large) constant c . Then the number of nodes of T whose depth is at most i that are intersected by any line ℓ is $O(r^{i(1/2+\epsilon)})$.*

Proof. Recall that each node of T corresponds to a triangle and T has the property that the number of children of any node intersected by any particular line ℓ is $O(r^{1/2} \log r)$.

Therefore, the number of nodes $x(i)$ of T at level i that intersect ℓ is given by the recurrence

$$x(i) \leq \begin{cases} 1 & \text{for } i = 0 \\ (cr^{1/2} \log r) \cdot x(i-1) & \text{for } i > 0 \end{cases}$$

which resolves to $(cr^{1/2} \log r)^i = O(r^{i(1/2+\epsilon)})$ for $r > (c \log r)^{1/\epsilon}$. \square

An i -set of a rooted tree T is a set of vertices in T all of which are at distance at most i from the root of T and in which no vertex in the set is the ancestor of any other vertex in the set. Note that if T is a partition tree defined in Section 3 then an i -set of T is a set of disjoint triangles. We say that a set of regions $X = \{X_1, \dots, X_m\}$, $X_i \subseteq \mathbb{R}^2$, is in k -general position if there is no line that intersects k or more elements of X .

Lemma 2. *Let $\epsilon > 0$, let T be the partition tree defined in Section 3 using a value $r > (c \log r)^{1/\epsilon}$ for some (sufficiently large) constant c , and let V be an i -set of T . Then V contains a subset $V' \subseteq V$ that is in k -general position and has size $\Omega(|V|/r^{i(1/2+\epsilon+4/k)})$.*

Proof. We will prove the lemma using the probabilistic method [2]. Let V' be a Bernoulli sample of V where each element is selected independently with probability $p = r^{-i(1/2+\epsilon+\delta)}$, where δ is a constant with $\delta > 4/k$. We will show that

$$\Pr \{V' \text{ is in } k\text{-general position and } |V'| = \Omega(|V|/r^{i(1/2+\epsilon+\delta)})\} > 0 ,$$

thus proving the existence of a set V' satisfying the conditions of the lemma.

Consider any line ℓ . By Lemma 1, ℓ intersects at most $cr^{i(1/2+\epsilon)}$ elements of V for some constant c . The probability that ℓ intersects k or more elements of V' is therefore no more than

$$\binom{cr^{i(1/2+\epsilon)}}{k} \cdot p^k \leq (cr^{i(1/2+\epsilon)} p)^k = c^k r^{ki(1/2+\epsilon) - ki(1/2+\epsilon+\delta)} = c^k r^{-ki\delta}$$

The nodes in V define a *test set* L of $O(|V|^2) = O(r^{i(4+\epsilon)})$ lines such that V' is in k -general position if and only if no line in L intersects k or more elements of V' . The probability that any line in L intersects more than k elements of V' is therefore at most $O(r^{i(4+\epsilon)} c^k r^{-ki\delta}) = O(c^k r^{i(4+\epsilon-k\delta)}) = o(1)$ for any constant $\delta > 4/k + \epsilon$.

The above argument shows that the nodes in V' are quite likely to be in k -general position. To see that V' is sufficiently large, we simply observe that $|V'|$ is a binomial($|V|, p$) random variable and therefore has median value at least $\lfloor p|V| \rfloor = \Omega(|V|/r^{i(1/2+\epsilon+\delta)})$. In particular, $\Pr\{|V'| \geq \lfloor p|V| \rfloor\} \geq 1/2$. Therefore,

$$\Pr \{V' \text{ is in } k\text{-general position and } |V'| = \Omega(|V|/r^{i(1/2+\epsilon+\delta)})\} \geq 1 - (o(1) + 1/2) > 0 .$$

Setting δ sufficiently close to (but larger than) $4/k + \epsilon$ completes the proof. \square

We are now ready to show that the search time in our data structure is a lower bound on the entropy of any Steiner triangulation of G . Recall that, by Theorem 3, the entropy of a minimum entropy Steiner triangulation of G is a lower bound on the expected cost of any linear decision tree for point location in G .

Lemma 3. *Let T be the partition tree defined in Section 3, let L denote the set of leaves of T , and let $H^* = H(\Delta^*)$ be the entropy of a Steiner triangulation Δ^* of G . Then $H^* = \Omega(H(L) - 1)$*

Proof. This proof mixes the ideas from the proofs of Lemma 3 by Dujmović *et al.* [16] and Lemma 4 by Collette *et al.* [12].

Let T' be the tree obtained from T by removing all terminal leaves, and let L' denote the set of leaves of T' . Note that L' is a Steiner triangulation of G and that

$$H(L') = H(L) - O(\log r) = H(L) - O(1)$$

since each triangle in L' is partitioned in $O(r^2)$ triangles in L .

Partition L' into groups G_1, G_2, \dots , where G_i contains all leaves v such that $1/2^{i-1} \geq \Pr(v) \geq 1/2^i$. Further partition each group G_i into subgroups $G_{i,1}, \dots, G_{i,t_i}$ with the property that each group $G_{i,j}$ with $j \in \{1, \dots, t_i - 1\}$ is in k -general position and has size at least $2^{\gamma i}$ for some constant $\gamma > 0$. Furthermore, the final group, G_{i,t_i} has size at most $O(2^{\beta i})$, for some constant $\beta < 1$. This partitioning is accomplished by repeatedly applying Lemma 2 to remove a subset $G_{i,j} \subseteq G_i$ that is in k -general position and has size $2^{\gamma i}$, stopping the process once the size of G_i drops below $2^{\beta i}$. This works provided that we choose β , k , and r so that $\beta > ((\log r)/(\log r - 1))(1/2 + \epsilon + 4/k)$ and set $\gamma = \beta - ((\log r)/(\log r - 1))(1/2 + \epsilon + 4/k)$.

Now, consider any Steiner triangulation Δ^* of G and let t be a triangle in Δ^* . Note that t cannot contain any triangle in L' since each element in L' is non-terminal in T and therefore its interior intersects at least two faces of G . Therefore, any subgroup $G_{i,j}$ intersected by t must intersect one of t 's three edges. Since each $G_{i,j}$ is in k -general position, this means that t intersects at most $3k$ elements of $G_{i,j}$. It follows [13, Lemma 3] that

$$H^* \geq H(L') - H(\{\cup G_{i,j} : i \in \mathbb{N}, j \in \{1, \dots, t_{i,j}\}\}) - O(1) .$$

Thus, all that remains is to upper-bound the contribution of $\bar{H} = H(\{\cup G_{i,j} : i \in \mathbb{N}, j \in \{1, \dots, t_{i,j}\}\})$.

$$\begin{aligned} \bar{H} &= H(\{\cup G_{i,j} : i \in \mathbb{N}, j \in \{1, \dots, t_{i,j}\}\}) \\ &= \sum_{i=1}^{\infty} \sum_{j=1}^{t_i} \Pr(\cup G_{i,j}) \log(1/\Pr(\cup G_{i,j})) \\ &= \sum_{i=1}^{\infty} \left(\sum_{j=1}^{t_i-1} \Pr(\cup G_{i,j}) \log(1/\Pr(\cup G_{i,j})) + \Pr(\cup G_{i,t_i}) \log(1/\Pr(\cup G_{i,t_i})) \right) \\ &\leq \sum_{i=1}^{\infty} \left(\sum_{j=1}^{t_i-1} \Pr(\cup G_{i,j}) \log(2^{i-\alpha i}) + i 2^{\beta i - i + 1} \right) \\ &\leq (1 - \alpha)H(L') + O(1) . \end{aligned}$$

Thus, we have

$$H^* \geq H(L') - \bar{H} - O(1) \geq \alpha H(L') - O(1) \geq \alpha H(L) - O(1) = \Omega(H(L) - 1)$$

as required. \square

Theorem 4. *Let G be a (possibly disconnected) planar subdivision of size n and let D be a probability measure over \mathbb{R}^2 . There exists a data structure T that, given G and D , can be constructed in $O(n)$ time, has $O(n)$ size, and can answer point location queries in G in $O(H^*)$ expected time, where H^* is the expected time to answer point location queries in G using any linear decision tree.*

Proof. The data structure is, of course, the partition tree T of Section 3 and some backup structure that can answer queries in $O(\log n)$ worst case time in case a query reaches a non-terminal leaf of T . The expected time answer queries in T is

$$\sum_{t \in L} \Pr(t) O(\text{depth}_T(t)) = \sum_{t \in L} \Pr(t) O(\log(1/\Pr(t))) = O(H(L)) .$$

On the other hand, by Lemma 3 and Theorem 3, the expected time required by any linear decision tree for answering queries in G is

$$H^* = \Omega(H(L) - 1) ,$$

which completes the proof. \square

We finish by observing that the tree T in Section 3 has sublinear size. Indeed, for any constant $0 \leq d \leq 1$, we can construct a tree T of size $O(n^d)$ that satisfies the conditions of Lemma 3. Thus, we can think of T as a sublinear sized filter that can take any point location structure with $O(\log n)$ worst-case query time and make it into a distribution-sensitive data structure. In particular, one can combine T with the succinct point location structure of Bose *et al.* [8, Theorem 2], to obtain the following result:

Theorem 5. *Let G be a (possibly disconnected) planar subdivision of size n and let D be a probability measure over \mathbb{R}^2 . There exists a data structure T that, given G and D , can be constructed in $O(n)$ time and can answer point location queries in G in $O(H^*)$ expected time, where H^* is the expected time to answer point location queries in G using any linear decision tree. This structure is represented as a permutation of the vertices of G and an additional $o(n)$ bits.*

References

- [1] U. Adamy and R. Seidel. On the exact worst case query complexity of planar point location. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 609–618, 1998.
- [2] N. Alon and J. H. Spencer. *The Probabilistic Method*. John Wiley & Sons, Hoboken, third edition, 2008.
- [3] S. Arya, S. W. Cheng, D. M. Mount, and H. Ramesh. Efficient expected-case algorithms for planar point location. In *Proceedings of the Seventh Scandinavian Workshop on Algorithm Theory*, pages 353–366, 2000.

-
- [4] S. Arya, T. Malamatos, and D. M. Mount. Nearly optimal expected-case planar point location. In *Proceedings of the 41st annual Symposium on Foundations of Computer Science*, pages 208–218, 2000.
 - [5] S. Arya, T. Malamatos, and D. M. Mount. Entropy-preserving cuttings and space-efficient planar point location. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 256–261, 2001.
 - [6] S. Arya, T. Malamatos, and D. M. Mount. A simple entropy-based algorithm for planar point location. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 262–268, 2001.
 - [7] S. Arya, T. Malamatos, D. M. Mount, and K. C. Wong. Optimal expected-case planar point location. *SIAM Journal on Computing*, 37(2):584–610, 2007.
 - [8] P. Bose, E. Chen, M. He, A. Maheshwari, and P. Morin. Succinct geometric indexes supporting point location. In *Proceedings of the 20th ACM-SIAM Symposium on Discrete Algorithms (SODA 2009)*, pages 635–644, 2009. Submitted to *ACM Transactions on Algorithms*, March 2009.
 - [9] T. M. Chan. Point location in $o(\log n)$ time, Voronoi diagrams in $o(n \log n)$ time, and other transdichotomous results in computational geometry. In *Proceedings of the 47th annual Symposium on Foundations of Computer Science*, pages 333–342, 2006.
 - [10] T. M. Chan and M. Pătraşcu. Transdichotomous results in computational geometry, i: Point location in sublogarithmic time. *SIAM Journal on Computing*, 39:703–729, 2009.
 - [11] B. Chazelle and E. Welzl. Quasi-optimal range searching in spaces of finite VC-dimension. *Discrete & Computational Geometry*, 4:467–489, 1989.
 - [12] S. Collette, V. Dujmović, J. Iacono, S. Langerman, and P. Morin. Distribution-sensitive point location in convex subdivisions. In *Proceedings of the 19th ACM-SIAM Symposium on Discrete Algorithms (SODA 2008)*, pages 912–921, 2008.
 - [13] S. Collette, V. Dujmović, J. Iacono, S. Langerman, and P. Morin. Entropy, triangulation, and point location in planar subdivisions. Technical Report cs0905.3584, arXiv, March 2009.
 - [14] L. Devroye. Bounds for the uniform deviation of empirical measures. *Journal of Multivariate Analysis*, 12:72–79, 1982.
 - [15] L. Devroye and G. Lugosi. *Combinatorial Methods in Density Estimation*. Springer-Verlag, New York, 2001.
 - [16] V. Dujmović, J. Howat, and P. Morin. Biased range trees. In *Proceedings of the 20th ACM-SIAM Symposium on Discrete Algorithms (SODA 2009)*, pages 486–495, 2009.
 - [17] H. Edelsbrunner, L. J. Guibas, and J. Stolfi. Optimal point location in a monotone subdivision. *SIAM Journal on Computing*, 15(2):317–340, 1986.
-

-
- [18] D. Haussler and E. Welzl. ε -nets and simplex range queries. *Discrete Computational Geometry*, 2:127–151, 1987.
- [19] J. Hershberger and S. Suri. A pedestrian approach to ray shooting: Shoot a ray, take a walk. *Journal of Algorithms*, 18(3):403–431, 1995.
- [20] J. Iacono. Optimal planar point location. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 240–241, 2001.
- [21] J. Iacono. Expected asymptotically optimal planar point location. *Computational Geometry Theory and Applications*, 29(1):19–22, 2004.
- [22] D. Kirkpatrick. Optimal search in planar subdivisions. *SIAM Journal on Computing*, 12(1):28–35, 1983.
- [23] J. Matoušek. Efficient partition trees. *Discrete & Computational Geometry*, 8(3):315–334, 1992.
- [24] K. Mulmuley. A fast planar partition algorithm. *Journal of Symbolic Computation*, 10:253–280, 1990.
- [25] M. Pătraşcu. Planar point location in sublogarithmic time. In *Proceedings of the 47th annual Symposium on Foundations of Computer Science*, pages 325–332, 2006.
- [26] N. Sarnak and R. E. Tarjan. Planar point location using persistent search trees. *Communications of the ACM*, 29(7):669–679, 1986.
- [27] N. Sauer. On the density of families of sets. *Journal of Combinatorial Theory Series A*, 13:145–147, 1972.
- [28] V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16:264–280, 1971.